Apple II
Technical Notes

Developer Technical Support

Apple IIGS
#36: Port Driver Specifications

Revised by: Matt Deatherage & Suki Lee September 1989
Written by: Dan Hitchens May 1988

This Technical Note describes how to write your own drivers for Apple IIGS ports.
Changed since January 1989: Added description of new port driver structure.

Introduction

A port driver handles certain hardware-specific duties for the Print Manager, such as initializing firmware and handling low-level hardware handshaking protocols, if any are implemented. The port driver structure, like the printer driver structure, insulates the Print Manager from low-level details of printers and interface cards (or ports) so that the same calls work across various hardware configurations, provided drivers are installed on the boot disk.

Note that a port driver could also easily be called a card driver; the term port is used because the first ones written were for the internal ports of the Apple IIGS. A port driver could interface any printer (for which there is a printer driver) with any kind of port or peripheral card that can handle it. A familiar example would be a parallel printer interface card—a port driver for a parallel card would enable the Print Manager to print graphics to any parallel printer connected to it (provided, again, there was a printer driver for the particular printer installed).

In general, you need a port driver for each port or interface card through which you intend to print, and a printer driver for each printer to which you intend to print. On System Disk 4.0, Apple provides port driver files for the printer port (PRINTER), the modem port (MODEM), a port connected to the AppleTalk network (APPLETALK), and a parallel printer interface card (PARALLEL.CARD). Apple also provides printer drivers for the ImageWriter and ImageWriter II (IMAGEWRITER), the ImageWriter LQ (IMAGEWRITER.LQ), the LaserWriter family.(LASERWRITER), and an Epson (EPSON). With this configuration, you can print to any of the printer types above through any of the ports, cards, or over AppleTalk. Other printer drivers and port drivers would extend the user's selection of available configurations.

What's in a Port Driver

File Structure

Users can install new port drivers into the system by copying a port driver file into a subdirectory called DRIVERS within the SYSTEM subdirectory or by running the Installer if the driver is supplied with a script to install it. The port driver file must be of type $BB. There are two kinds of port drivers: local drivers, intended to drive a printer connected locally, and network drivers, which handle printers connected over an AppleTalk network. Local drivers have an auxiliary type of $0002, and AppleTalk drivers (there should be only one) have an auxiliary type of $0003.

Port Driver Calls

A port driver must support the following calls:

`PrDevPrChanged`	$1913	
`PrDevStartup`	$1A13	
`PrDevShutDown`	$1B13	
`PrDevOpen`	$1C13	
`PrDevRead`	$1D13	
`PrDevWrite`	$1E13	
`PrDevClose`	$1F13	
`PrDevStatus`	$2013	
`PrDevAsyncRead`	$2113	(alias `PrDevInitBack`)
`PrDevWriteBackground`	$2213	(alias `PrDevFillBack`)
`PrPortVer`	$2413	
`PrDevIsItSafe`	$3013	

Note that a network port driver has much more work to do than a regular (local) port or card driver. A local driver only has to worry about one printer, whereas a network port driver may find that there is not even a printer available on a running network. The information on network drivers is provided mostly for informational purposes; you should never find it necessary to write your own AppleTalk port driver.

Entering and Exiting a Port Driver

Entering and exiting is the same as described for the printer driver calls in Apple IIGS Technical Note #35, Printer Driver Specifications. The new driver structure described there applies as well. As of this writing, there are no optional calls a port driver may support. The documented list must be supported in its entirety.

PrDevPrChanged **$1913**

Description:

The Print Manager makes this call every time the user accepts this port driver in the Choose Printer dialog.

Input:	LONG	printer name pointer

Direct Connect:

- Makes sure that this port has been set up correctly in the Control Panel (parity, baud rate, etc.), and puts up an alert for the user if it has not been. Remember that if you change settings, even at the user's request, you should change the Battery RAM parameters as well, so the setting changes will be reflected when the user enters the Control Panel.

Network:

- Copies the printer name to local storage for use in the `NBPLookup` function of the AppleTalk `PAPopen` and `PAPstatus` calls, usually by placing it in the AppleTalk parameter block. This function is similar to that performed by `PrStartUp`, except that `PrDevPrChanged` is called whenever the printer is changed by the user with the Choose Printer dialog.

PrDevStartUp **$1A13**

Description:

This call is not required to do anything. However, if your driver needs to initialize itself by allocating memory or other setup tasks, this is the place to do it. Network drivers should copy the printer name to a local storage area for later use.

Input:	LONG	printer name pointer
	LONG	zone name pointer

Direct Connect:

- Required to do nothing. This is a good place to do your own set-up tasks, if you have any.

Network:

- Copies the printer name and the zone name to local storage for use in the `NBPLookup` function of the AppleTalk `PAPopen` and `PAPstatus` calls, usually by placing it in the AppleTalk parameter block.

PrDevShutDown **$1B13**

Description:

This call, like `PrDevStartUp`, is not required to do anything. However, if your driver performs other tasks when it starts, from the normal (allocating memory) to the obscure (installing heartbeat tasks), it should undo them here. If you allocate anything when you

start, you should deallocate it when you shutdown. Note that this call may be made without a balancing `PrDevStartUp`, so be prepared for this instance. For example, do not try to blindly deallocate a handle that your `PrDevStartUp` routine allocates and stores in local storage; if you have not called `PrDevStartUp`, there is no telling what will be in your local storage area.

Input: none

PrDevOpen $1C13

Description:

This call basically prepares the firmware for printing. It must initialize the firmware for both input and output. Input is required so the connected printer may be polled for its status.

A network driver has considerably more work to do, including the possibility of asynchronous communications. Details are provided below.

Input:		
	LONG	completion routine pointer
	LONG	reserved long

Direct Connect:

- Initializes the firmware for input and output, preparing for reading from or writing to the printer.
- If the completion pointer is `NIL`, then `RTL`. If it is not `NIL`, then perform a `JSL` to the completion routine.

Network:

- Initializes the `End-Of-Write` parameter in the AppleTalk `PAPWrite` parameter block to zero. Never call AppleTalk `INIT` to initialize the firmware.
- If the completion pointer is `NIL`, then prepares for synchronous communications. If it is not `NIL`, prepares for asynchronous printing.
- Calls AppleTalk `PAPopen` to make connection, returning an error if one is returned to you.
- Stores the AppleTalk Session number in the `PAPRead`, `PAPWrite` and `PAPClose` parameter blocks.
- Executes an `RTL` if there is no completion routine (pointer is `NIL`), otherwise perform a `JSL` to the completion routine.

PrDevRead $1D13

Description:

This call reads input from the printer.

Input:		
	WORD	space for result
	LONG	buffer pointer
	WORD	number of bytes to transfer

Output: WORD number of bytes transferred

Direct Connect:
- Reads a specified number of bytes from the printer into the buffer.

Network:
- Calls AppleTalk `PAPRead` to read synchronously. Since there is no completion pointer, reading from a network device must always be done synchronously. To read asynchronously, use `PrDevAsyncRead`.

PrDevWrite **$1E13**

Description:

Writes the data in the buffer to the printer and calls the completion routine.

Input:		
	LONG	write completion pointer
	LONG	buffer pointer
	WORD	buffer length

Direct Connect:

- Writes the contents of the buffer to the printer.
- If the completion pointer is `NIL`, then `RTL`. If it is not, then perform a `JSL` to the completion routine.

Network:

- If the completion pointer is `NIL`, then writing will occur synchronously. Otherwise, writing will occur asynchronously.
- Calls AppleTalk `PAPWrite` to transfer the contents of the buffer.
- If the completion pointer is `NIL`, then `RTL` to the caller. Otherwise, perform a `JSL` to the completion routine first, with the error code in the accumulator.

PrDevClose **$1F13**

Description:

This call is not required to do anything. However, if you allocate any system resources with `PrDevOpen`, you should deallocate them at this time. As with start and shutdown, note that `PrDevClose` could be called without a balancing `PrDevOpen` (the reverse is not true), and you must be prepared for this if you try to deallocate resources which were never allocated.

Input: none

Direct Connect:

- No required function.

Network:

- Sets `End-Of-Write` parameter in AppleTalk `PAPWrite` parameter block to one.
- Calls `PAPWrite` with no data.
- Calls `PAPClose`.

PrDevStatus **$2013**

Description:

This call performs differently for direct connect and network drivers. For direct connect drivers, it currently has no required function, although it may return the status of the port in the future. For network drivers, it calls an AppleTalk status routine, which returns a status string in the buffer (normally a string like "Status: The print server is spooling your document").

Input:	LONG	status buffer pointer

Direct Connect:

- Does nothing.

Network:

- Calls AppleTalk `PAPStatus`.

PrDevAsyncRead **$2113**

Description:

Since `PrDevRead` cannot read asynchronously, this call is provided for that task. Note that this does nothing for direct connect drivers, and if the completion pointer is `NIL`, it behaves for network drivers exactly as `PrDevRead` does.

Input:	WORD	space for result
	LONG	completion pointer
	WORD	buffer length
	LONG	buffer pointer
Output:	WORD	number of bytes transferred

Direct Connect:

- Does nothing.

Network:

- If the completion pointer is `NIL`, then performs exactly as `PrDevRead`.
- Calls AppleTalk `PAPRead`; the actual length read is passed back in the `PAPRead` parameter block.
- Perform a `JSL` to the completion routine, which returns the length read in the `X` register and an EOF flag in the `Y` register. As usual, the accumulator contains the error code and the carry is set if an error occurs.
- In the case of a synchronous call, it performs a `JSL` to the completion routine, which pushes the length read onto the stack.

PrDevWriteBackground **$2213**

Description:

This routine is not implemented at this time.

Input:		
	LONG	completion procedure pointer
	WORD	buffer length
	LONG	buffer pointer

PrPortVer **$2413**

Description:
Returns the version number of the currently installed port driver.

Input:	WORD	space for result
Output:	WORD	Port driver's version number

Direct Connect and Network:
- Gets the internal version number of the port driver and returns it on the stack.

Note:
The internal version number is stored as a major byte and a minor byte (i.e., $0103 represents version 1.3)

PrDevIsItSafe **$3013**

Description:
This call checks to see if the port or card which your driver controls is enabled. It should check at least the corresponding bit of $E0C02D, and checking the Battery RAM settings wouldn't hurt any either.

Input:	WORD	space for result
Output:	WORD	Boolean indicating if port is enabled

Direct Connect and Network:
- Checks the system to see if the hardware and/or firmware for the card or port this driver controls is enabled, and returns TRUE if it is safe to proceed and FALSE if not. Note that for a port driver that controls an interface card, this call should return FALSE if the card is disabled and the port is enabled, while for a port driver which controls an Apple IIGS internal port, the returned value should be TRUE if the port is enabled and FALSE if not.

Further Reference

- *Apple IIGS Toolbox Reference*, Volumes 1 & 2
- Apple IIGS Technical Note #35, Printer Driver Specifications